FlyAway (An Airline Booking Portal)

Developed by Tata Sai Vineeth

- ## Sprints planned and the tasks achieved in them

  Flyaway was divided in 3 sprints of 1 week each.

Sprint 1: -

- Initialized the project, made it Hibernate project and added dependencies: - MySql connector, Hibernate libraries, Servlet API.
- Setting MySQL database environment and tomcat server
- Made classes for entities and established relationship between them.
- Configured hibernate and mapped classes as entities.
- Populated the data by hibernate

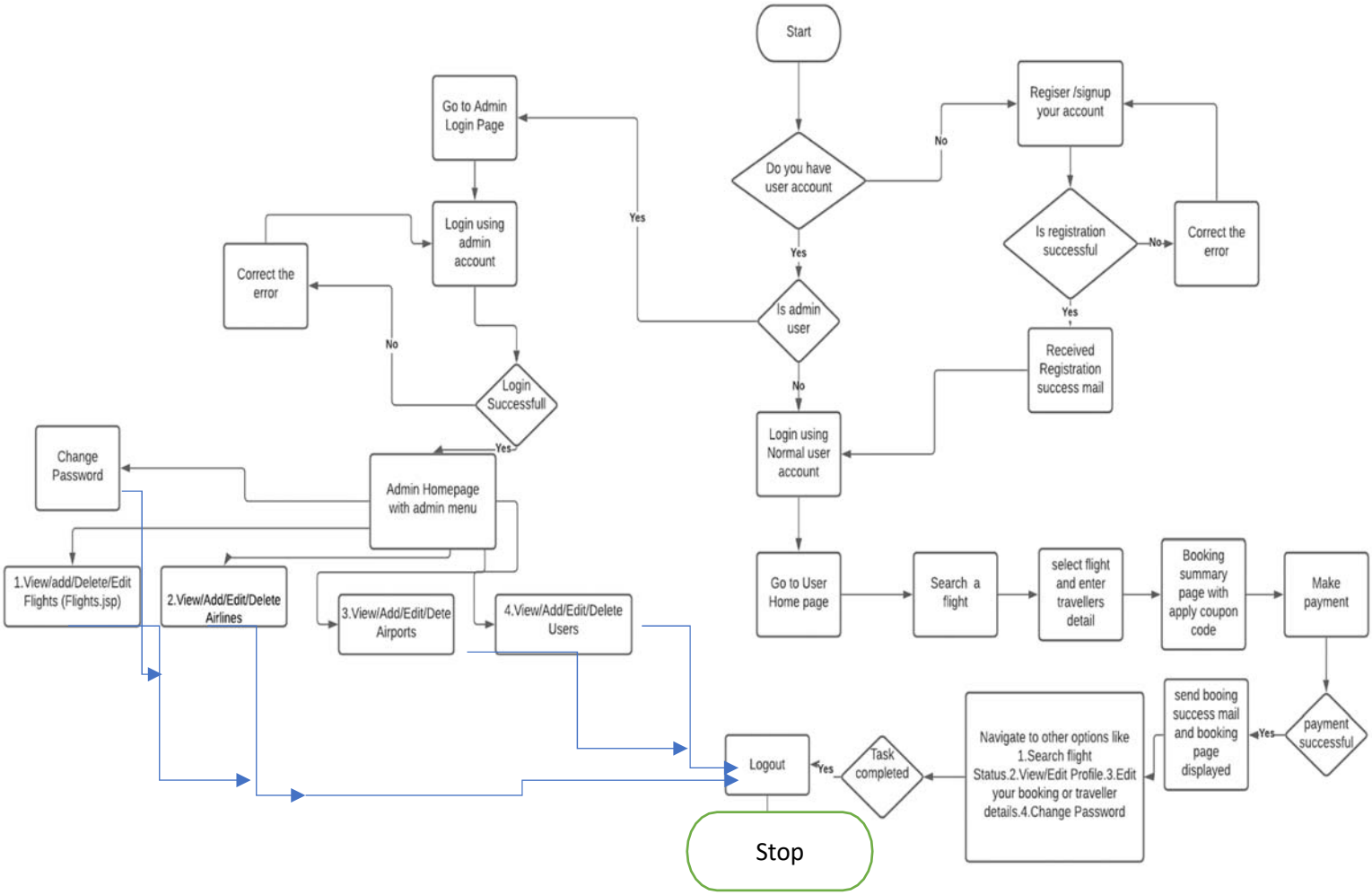
Sprint 2: -

- Planned the structure of the website.
- Made a Hibernate util. class to use it across the servlets.
- Made servlets for each class list and set up navigation link for them.

Sprint 3: -

- Made the master list HTML for master list of places for source and destination, master list ofairlines.
- Also Made A list of flights where each flight has a source, destination, airline, and ticket price.
- Set up Servlets for each of the master list to send data as a list.

- **Algorithms and flowcharts of the application**



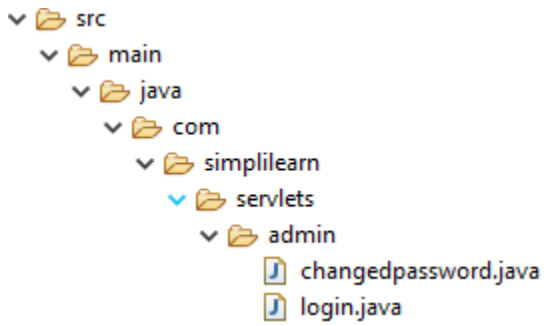
Core concepts used in the project

- Server request response object
- HTTP Session, Cookies URL Redirect Hidden Field.
- Collection Framework
- DateTime object with simpleDateFormat
- Hibernate JPA, Different types of mapping (one to one, one to many, many to one, many to many).
- Sorting technique using database query as well as using Collection framework comparator methods.
- MVC Framework where View (Html pages) , Controller (Servlet pages), Data Model (Entity classes and interface which interact to backend database using hibernate framework)
- Table Joins, Filters, Subqueries Transaction Control
- HTML to design view
- Web container such as Tomcat server usage deployment of web application and running/configuration.
- MySQL Database and SQL Query language.

Classes: -

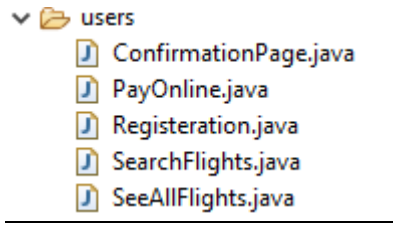
There were 7 entity classes named Flight Booking, Traveler, Airline, Airport, User, Admin and Classes. Which all are inside java layer i.e. com.simplilearn package.

Com.simplilearn.admin:

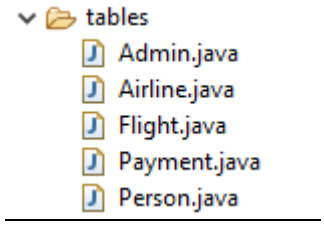


On top of dto layer we have DAO layer which provides interface to access database CRUD operation as well implements interface using Hibernate Session, Session factory and Transaction objects.

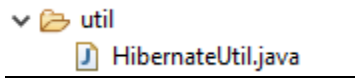
Com.simplilearn.users:



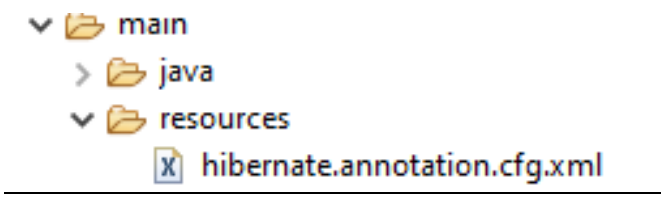
Com.simplilearn.tables:



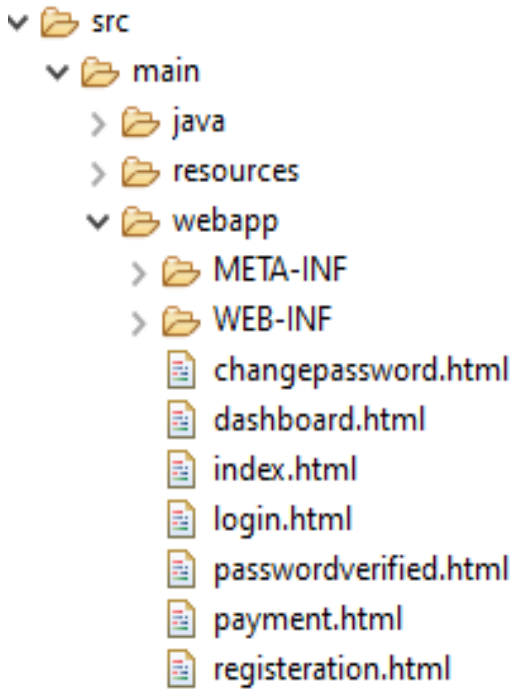
Com.simplilearn.util:



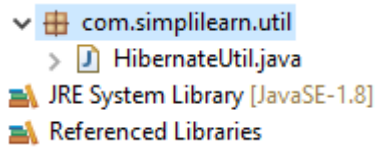
Main.resourses:



At top most layer (View) we have web content like JSP/HTML/CSS and other files through which user interacts and sends request or receives response.



Com.simplilearn package to send mail and for some helper classes like format date etc



In the hibernate.cfg.xml we have the entire configuration related to database connectivity as well asthe resource class defined. As we have used annotated class so each entity class has annotation related to database entities like table, column and mapping.

hibernate.annotation.cfg.xml

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE hibernate-configuration PUBLIC
        "-//Hibernate/Hibernate Configuration DTD 3.0//EN"
        "http://www.hibernate.org/dtd/hibernate-configuration-3.0.dtd">

<hibernate-configuration>
    <session-factory>
        <property name="hibernate.connection.driver_class">com.mysql.cj.jdbc.Driver</property>
        <property name="hibernate.connection.url">jdbc:mysql://localhost:3306/flyawaydb</property>
        <property name="hibernate.connection.username">root</property>
        <property name="hibernate.connection.password">12331a0299@Sai</property>
        <property name="hibernate.connection.pool_size">1</property>
        <property name="hibernate.current_session_context_class">thread</property>
        <property name="hibernate.show_sql">true</property>
        <property name="hibernate.dialect">org.hibernate.dialect.MySQL5Dialect</property>
        <property name="hibernate.id.new_generator_mappings">false</property>

        <mapping class="com.simplilearn.tables.Admin" />
        <mapping class="com.simplilearn.tables.Airline" />
        <mapping class="com.simplilearn.tables.Flight" />
        <mapping class="com.simplilearn.tables.Payment" />
        <mapping class="com.simplilearn.tables.Person" />
    </session-factory>
</hibernate-configuration>
```

Servlets: -

- **LoginServlet :**- to Authenticate a user login using Login.java. If Authentication failes user gets appropriate error message. If successful set the Session object and redirect User to Home.java page

- **AdminLoginServlet** (Uses AdminDAO interface): - to Authenticate a Admin login using LoginAdmin.java. If Authentication fails user gets appropriate error message. If successful set the Session object and redirect admin to HomeAdmin.java page

- **ChangePassword** Servlet:-to Change the password for admin and User it handles user request from changePassword modal dialog.

- **AirlineServlet** (Used AirlineDAO interface): - to Mange airlines (Add/Edit/Delete/View airlines) by admin users. Note If a airline is deleted it also deletes dependent flight there is one to many relationship between airline and flight. Mapped with Airlines.java , and used by Flights.java

- **FlightServlet** (Uses FlightDAO):- to Manage flight (Add/Edit/Delete/View) admin uses Flights.java. User can also check flight status using FlightStatus.java.

HTML file: -

Main.html is a file for the entire sheet across the website embedded in header.java so it can be included in entire website. To accelerate design process we used popular html framework. Here we used 7 kinds of .html

1. Change password.html: By using this, we can change password in the server. .
2. Dashboard.html: By using this, we can view home, logout and change password buttons.
3. Index.html: By using this, we can search for flights on server.
4. Login.html: By using this, we can enter the credentials of the website.
5. Passwordverified.html: By using this, we can verify the password.
6. Payment.html: By using this, we can view payment window on screen.
7. Registration.html: By using this, we can register with our name, mobile number and emailid.

Note: - After every class push the code to git hub git hub link is –

https://github.com/Sai153793/flyaway

If any issue please feel free to contact me my details are-

Thanks
Develop by
Tata Sai Vineeth
Email id – t.saivineeth111@gmail.com
Mobile No- 9000665438